310, and/or the example batch execution controller **312**. The processor **812** of the illustrated example is in communication with a main memory including a volatile memory **814** and a non-volatile memory **816** via a bus **818**. The volatile memory **814** may be implemented by Synchronous Dynamic Random Access Memory (SDRAM), Dynamic Random Access Memory (DRAM), RAMBUS Dynamic Random Access Memory (RDRAM) and/or any other type of random access memory device. The non-volatile memory **816** may be implemented by flash memory and/or any other desired type of memory device. Access to the main memory **814**, **816** is controlled by a memory controller.

[0059] The processor platform **800** of the illustrated example also includes an interface circuit **820**. The interface circuit **820** may be implemented by any type of interface standard, such as an Ethernet interface, a universal serial bus (USB), and/or a PCI express interface.

[0060] In the illustrated example, one or more input devices **822** are connected to the interface circuit **820**. The input device(s) **822** permit(s) a user to enter data and commands into the processor **812**. The input device(s) can be implemented by, for example, an audio sensor, a microphone, a camera (still or video), a keyboard, a button, a mouse, a touchscreen, a track-pad, a trackball, isopoint and/or a voice recognition system.

[0061] One or more output devices **824** are also connected to the interface circuit **820** of the illustrated example. The output devices **824** can be implemented, for example, by display devices (e.g., a light emitting diode (LED), an organic light emitting diode (OLED), a liquid crystal display, a cathode ray tube display (CRT), a touchscreen, a tactile output device, a light emitting diode (LED), a printer and/or speakers). The interface circuit **820** of the illustrated example, thus, typically includes a graphics driver card, a graphics driver chip or a graphics driver processor.

[0062] The interface circuit **820** of the illustrated example also includes a communication device such as a transmitter, a receiver, a transceiver, a modem and/or network interface card to facilitate exchange of data with external machines (e.g., computing devices of any kind) via a network **826** (e.g., an Ethernet connection, a digital subscriber line (DSL), a telephone line, coaxial cable, a cellular telephone system, etc.).

[0063] The processor platform **800** of the illustrated example also includes one or more mass storage devices **828** for storing software and/or data. For example, the mass storage device **828** may include the example continuous historian database **320**, the example stage definitions database **314**, the example stage model database **316**, and/or the example batch recipe database **318** of FIG. **3**. Examples of such mass storage devices **828** include floppy disk drives, hard drive disks, compact disk drives, Blu-ray disk drives, RAID systems, and digital versatile disk (DVD) drives.

[0064] Coded instructions **832** to implement the methods of FIGS. **4-7** may be stored in the mass storage device **828**, in the volatile memory **814**, in the non-volatile memory **816**, and/or on a removable tangible computer readable storage medium such as a CD or DVD.

[0065] From the foregoing, it will be appreciated that the above disclosed methods, apparatus and articles of manufacture enable more accurate identification of process stages corresponding to particular process dynamics or interactions of process parameters throughout a batch process over known approaches that tie stages to batch events. As a result, the teachings disclosed herein enable more accurate identification of relevant process control data for generating models to apply for batch processes because the process control data occurring between the beginning and ending of a particular stage can be specifically isolated. Additionally, the precise beginning and ending of a stage in a currently executed batch process can be more precisely identified to determine the proper time to apply the models. In this manner, the batch control analytics can provide more accurate and reliable feedback and analysis on a current batch process than other known methods.

[0066] Although certain example methods, apparatus and articles of manufacture have been disclosed herein, the scope of coverage of this patent is not limited thereto. On the contrary, this patent covers all methods, apparatus and articles of manufacture fairly falling within the scope of the claims of this patent.

What is claimed is:

1. A method comprising:

determining, with a processor, a current stage in a current batch process based on a current value of a batch stage parameter, the current value of the batch stage parameter determined based on process control data associated with process parameters in the current batch process, the current stage determined independent of batch events defined by at least one of a start or an end of procedures, unit procedures, operations, or phases in a batch recipe; and

applying, with the processor, a model to the current batch process, the model corresponding to the current stage.

2. The method of claim **1**, wherein the current value of the batch stage parameter is defined based on user-specified definitions associated with conditions of the process parameters without reference to the batch events.

3. The method of claim **1**, wherein the current stage is defined by operator input received between successive ones of the batch events during runtime of the batch process, wherein the operator input satisfies a user-specified rule to define the current value of the batch stage parameter.

4. The method of claim **1**, wherein the current stage corresponds to a portion of the current batch process associated with a particular interaction of the process parameters.

5. The method of claim **1**, further including generating the model for the current stage of the current batch process based on a corresponding stage in each of multiple previously executed batch processes.

6. The method of claim **5**, wherein the model is generated by:

retrieving the process control data associated with the previously executed batch processes stored in a continuous historian database;

detecting a beginning of the corresponding stage in each previously executed batch process based on when the process control data associated with each previously executed batch process first satisfies user-specified definitions defining the current value of the batch stage parameter;

detecting an ending of the corresponding stage in each previously executed batch process based on when the process control data associated with each previously executed batch process no longer satisfies the user-specified definitions; and

analyzing the process control data associated with the previously executed batch processes between the